Professional and Scientific Staff Management, Inc.
Temporary Staff Fulfillment System
Deployment Proposal
2016-12-07

Rob Sanchez and Juan Carcamo
Knowledge Machines Analysis & Design

CIS 641, Fall 2016
Schymik

Contents

1. Introduction

The purpose of this document is to present a deployment proposal for a Temporary Staffing Fulfillment System (*TechStaff*) at Professional and Scientific Staff Management, Inc. of Toledo, OH. PSSM is a medium-sized, niche-market staffing agency operating in the Midwestern United States. It specializes in high-skilled technical and scientific temporary staffing placements and has experienced an unexpected surge in demand for suitable temporary staffing candidates from companies throughout the region in the last three years.

PSSM's current placement and arrangements system, such as it is, hasn't been able to keep up with the agency's growth and suffers from significant performance and usability problems. As a result, leadership has decided to replace it. They reached out to our firm, *Knowledge Machines Analysis & Design*, for a re-implementation of their solution.

PSSM's core competency is negotiating contracts and maintaining relationships between their counter-parties. They employ a minimal IT team with no plans to hire new staff. Their current staffing system consists of a set of related spreadsheets stored on the company's shared network drive. This arrangement does not support a high level of concurrency and suffers from a high rate of errors. It also doesn't protect client or candidate staff personal data to the satisfaction of the PSSM legal department.

PSSM are looking for a hosted, cloud-based web application to access from their offices in downtown Toledo, with physical data management to be handled offsite by a third party. Basic web application configuration access should be available to PSSM IT staff, but code development and maintenance is to be managed by KMAD through a standard support contract.

2. Description

2.1 Requirements

The main business requirement the *TechStaff* system meets is that of Staffing Request Fulfillment. In overview, one of PSSM's clients will phone or email and demand a temporary staffing placement, which PSSM attempts to fill by locating potential candidates. A set of criteria are given by the client and PSSM matches them to the availability pool.

PSSM's existing spreadsheet data is to be migrated into the *TechStaff* database. This process is relatively straight-forward: currently three spreadsheets are used: Contracts, Employees, and Requests. These more or less already fit the data model required. Each spreadsheet will represent a table in the TechStaff database, with a web application on top to manage the data and help PSSM execute their core business processes.

2.2 Business Processes

After a number of discussions between PSSM and KMAD, it was determined that Staffing Request Fulfillment consists of three main business processes: 1) Open a staffing request, 2) Place a candidate, and 3) Fill the staffing request. The people involved (in SAD parlance, "actors") are the PSSM Contract Manager, the PSSM Placement and Arrangements departments, and Temporary Employees in the candidate pool.

For each business process, we've functionally modeled a use-case and associated use-case description, which describe the interactions between actors and the system to achieve the goal of executing the business processes. These are detailed in sections 4.1.1 and 4.1.3 in the Functional Analysis section below. In addition, we've detailed the overall process flow for the fulfillment process in an Activity Diagram (section 4.1.2). We use these functional analysis design tools ("artifacts") to build the structural models of the system (section 4.2).

3. Effort Estimation

In order to estimate the level of effort required to deliver the solution, we've assembled a use case points analysis to estimate effort in person-hours. A use case points analysis operates on 4 main categories: 1) Actors, 2) Use Cases, 3) Technical Complexity, and 4) Environmental Factors. We use a standard use case points analysis spreadsheet model to determine the estimate.

3.1 Actors

Actors are counted and weighted according to their complexity. This results in an Unadjusted Actor Weight Total (UAW). The TechStaff application includes 1 average actor (the database) and 4 complex actor types (the endusers).

Unadjusted Actor Weight Total (UAW): [14]

3.2 Use Cases

Use cases are also counted and weighted according to their complexity. This results in an Unadjusted Use Case Weight Total (UUCW). The TechStaff application includes 3 average use cases (4-7 transactions each).

Unadjusted Use Case Weight Total (UUCW): [30]

3.3 Unadjusted Use-Case Points

The UAW and UUCW values are added together to yield UUCP, or Unadjusted Use-Case Points. This is used in the following Technical Complexity and Environmental Factors sections.

Unadjusted Use-Case Points (UUCP) = UAW + UUCW: 14 + 30 = [44]

3.4 Technical Complexity

The technical complexity of the project is estimated by considering and weighting a collection of technical factors, such as response time expectations, ease of use, portability, and concurrency. In particular, PSSM's solution requires more concurrency than currently available, because they have multiple staff members from Contracts, Placements and Arrangements requiring use of the system at the same time. This part of the analysis yields a Technical Factor Value (TFactor). This in turn is converted into a Technical Complexity Factor measurement according to an established industry formula.

Technical Factor Value (TFactor): [34.5]

Technical Complexity Factor (TCF) = 0.6 + (0.01 * TFactor): [0.945] = 0.6 + (0.01 *34.5)

3.5 Environmental Factors

Other environmental factors are considered to determine the effort estimate of the project. Criteria such as familiarity with the system development process, object-oriented experience, stability of requirements, and the whether part-time staff are used play a role. After quantifying their impact, this part of the analysis yields an Environmental Factor Value (EFactor). This in turn is converted into an Environmental Factor measurement according to an established industry formula.

Environmental Factor Value (EFactor): [25]

Environmental Factor (EF) = 1.4 + (-0.03 * EFactor): [0.65] = 1.4 + (-0.03 * 25.5)

3.6 UCP and PHM

With these data we can now calculate Adjusted Use Case Points (UCP) as well as the Person-Hours Multiplier (PHM). UCP adjusts our raw points estimate from step 3.3 for Technical and Environmental factors, and the PHM is used to estimate the number of person-hours required to build the system. Depending on the number of environmental factors score, this value could be 20 or 28. For PSSM TechStaff, the value is 20.

Adjusted Use Case Points (UCP) = UUCP * TCF * ECF: [27.027] = 44 * 0.945 * 0.65

Effort in person-hours = UCP * PHM: [540.54] = 20 * 27.027

3.7 Final Effort Estimate

After this set of calculations, we can estimate that it will take just over 540 hours to build the *TechStaff* solution for PSSM.

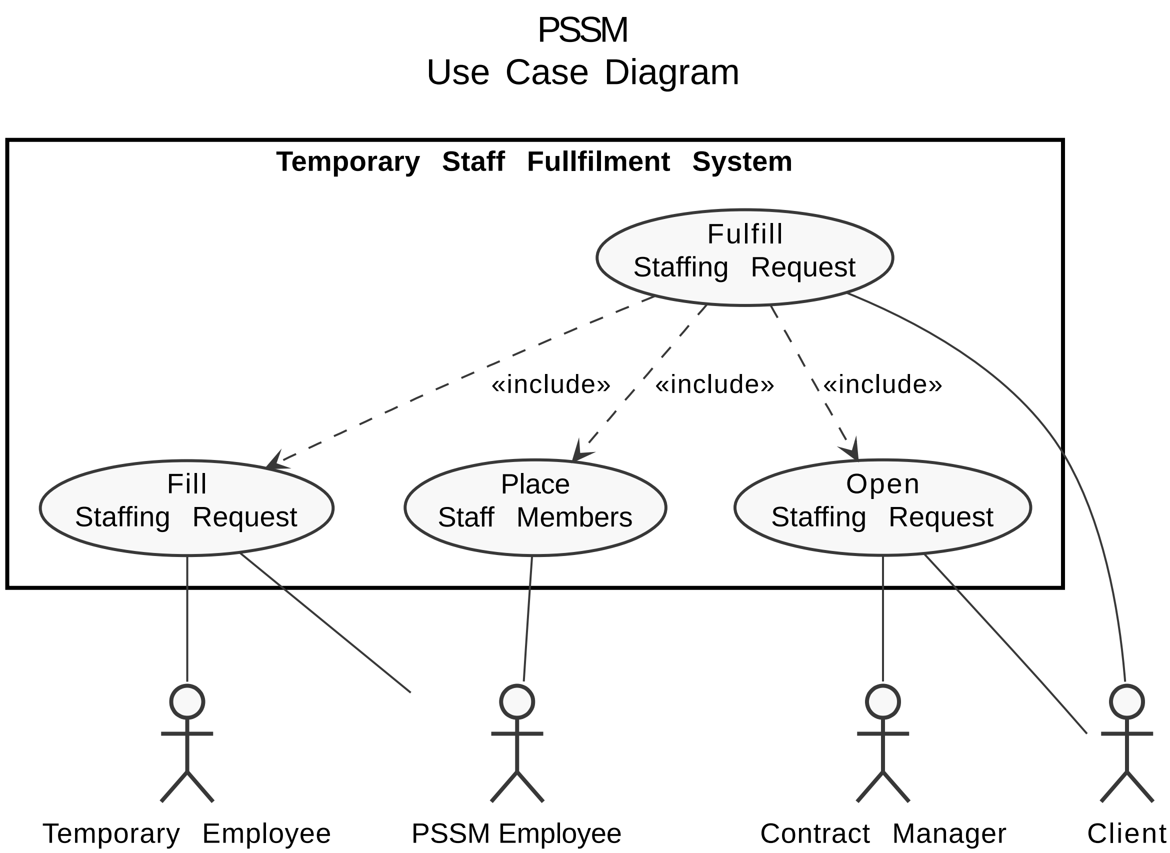
Effort in person-hours: [540. 54 hours]

4. Analysis

4.1 Functional

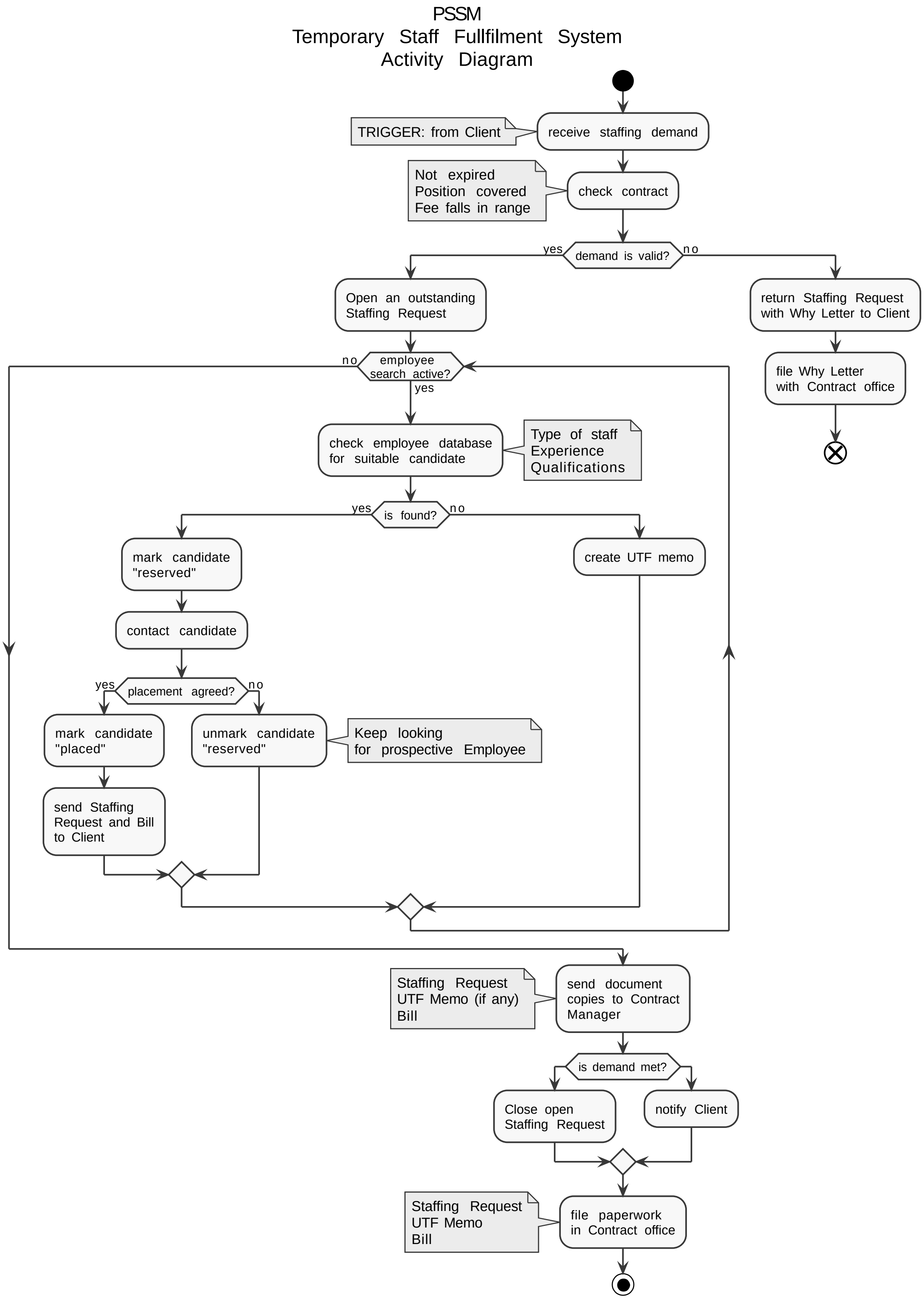
4.1.1 Use Case Diagram

The following Use Case Diagram depicts the main business processes performed required by the *TechStaff* system.



4.1.2 Activity Diagram

The following Activity Diagram



4.1.3 Use Case Descriptions

The following four Use Case Descriptions functionally document the business processes depicted in the Use Case Diagram and Activity Diagram presented in sections 4.1.1 and 4.1.2. These are used to build the structural models for the Staffing System.

Use Case Name:	Open Staffing Request (detailed)
Scenario:	Create a new outstanding Staffing Request.
Triggering event:	Client contacts PSSM and demands a new temporary staff arrangement.
Brief description:	PSSM receives a new demand for temporary staffing from the Client. PSSM's Contract Manager reviews the contract terms. If the new demand is not valid, the demand is returned to the Client with a Why Letter explaining. Otherwise, a new Staffing Request is opened in the Contract database and Placement is notified.
Actors:	Client, Contract Manager
Related use cases:	Place Staff Members, Fulfill Staffing Request
Stakeholders:	Contract Manager, Temporary Employee, Placement department
Pre-conditions:	A valid demand.
Post-conditions:	A new outstanding Staffing Request is created and the Placement department is notified.

Flow of activities:		
	Actor	System
	1. Client demands a new temporary staff arrangement 2. Contract manager reviews the client contract 3. Contract manager opens a new Staffing Request	2-1. Referencing the contract number on the new demand, the System pulls the Client contract details from the Contract database. 2-2. System creates a New Vehicle entry and saves associated details: serial number, description, make, model, features, etc. 3-1. System creates a new Staffing Request, marking it outstanding 3-2. System notifies Placement department of outstanding Staffing Request.

Exception conditions:	2-1. Demand is invalid, send Why Letter to Client explaining. File letter locally.

Use Case Name:	Place Staff Members (detailed)
Scenario:	Find suitable temporary staff to fill a client demand.
Triggering event:	Contract Manager opens a new outstanding Staffing Request, and the System notifies the Placement department.
Brief description:	Placement department checks position type, experience, and qualifications specified on the Staffing Request in the staff database.
Actors:	Placement department: PSSM Employee
Related use cases:	Open Staffing Request, Fill Staffing Request, Fulfill Staffing Request
Stakeholders:	Placement department, Contract Manager, Arrangements department, Temporary Employee
Pre-conditions:	A new outstanding Staffing Request must exist. A qualified Temporary Employee must be available.
Post-conditions:	A qualified Temporary Employee is marked as "reserved" and the Arrangements department is notified.

Flow of activities:		
	Actor	System
	1. A PSSM Employee from Placement checks the position type, experience, and qualifications specified in the Staffing Request against the database of available Temporary Employees. 2. The Temporary Employee is marked as "reserved"	1-1. System pulls available Temporary Employees matching the Staffing Request. 2-1. The System forwards the Staffing Request to the Arrangements department and notifies them that a possible match was found.

Exception conditions:	2-1. A qualified Temporary Employee is not available and an "unable-to-fill" Memo is explaining is attached to the Staffing Request.

Use Case Name:	Fill Staffing Request (detailed)
Scenario:	Arrange a Temporary Employee assignment for a Client.
Triggering event:	A possible qualified and available staff candidate is marked as “reserved” in the System, and Arrangements department is notified.
Brief description:	A potential temporary staff candidate is contacted, and after settling details and negotiating terms, the Temporary Employee is placed with the client and the Staffing Request is Closed.
Actors:	Arrangements department: PSSM Employee, Temporary Employee
Related use cases:	Place Staff Members, Fulfill Staffing Request
Stakeholders:	Client, Temporary Employee, Arrangements department
Pre-conditions:	A possible qualified and available temporary staff candidate has been marked as “reserved” in the System.
Post-conditions:	A Temporary Employee is marked as “placed” in the System. The open Staffing Request is closed. A copy of the original Staffing Request and a Bill is sent to the Client. The Staffing Request, any “unable-to-fill” memos and a copy of the Bill are sent to the Contract Manager.

Flow of activities:		
	Actor	System
	1. A PSSM Arrangements Employee contacts the temporary staff candidate. 2. The PSSM Employee settles any details and negotiates terms with the candidate. 3. The PSSM Employee places the Temporary Employee with the Client. 4. The PSSM Employee closes the open Staffing Request.	3-1. The System marks the Temporary Employee as “placed”. 4-1. The System closes the open Staffing Request. 4-2. Copy of the Staffing Request and a Bill are sent to the client. 4-3. The Staffing Request, a copy of the Bill, and any “unable-to-fill” Memo is sent to the Contract Manager.

Exception conditions:	3-1. Staffing Request not filled, client is notified. Staffing Request, Bill, and “unable-to-fill” Memo are filed in the contract office.

Use Case Name:	Fulfill Staffing Request (overview)
Scenario:	Open and fill a Staffing Request in order to place a Temporary Employee with a Client to meet a demand.
Triggering event:	A Client contacts PSSM to demand a new temporary staff arrangement.
Brief description:	The PSSM Contract Manager receives the Client demand and reviews the contract details. The Placement department finds a suitable staff member to meet the Client demand. Arrangements department negotiates the arrangement and places the employee with the Client.
Actors:	Client
Related use cases:	Open Staffing Request, Place Staff Members, Fill Staffing Request
Stakeholders:	Client
Pre-conditions:	Client and PSSM have an active contract in place.
Post-conditions:	A Temporary Employee is placed with the Client.

Flow of activities:		
	Actor	System
	1. Contract Manager opens a Staffing Request. 2. The Placement department identifies a suitable staffing candidate. 3. The Arrangements department negotiates with and places the Temporary Employee with the Client.	1. Placement department is notified. 2. Arrangements department is notified. 3. The Client is billed.

Exception conditions:	2-1. The Staffing Request cannot be fulfilled, the client is notified. The SR, Bill, and “unable-to-fill” Memo are filed in the Contract office.

4.2

5. Design

   The system will be named *TechStaff.*

6. User Interface

7. Physical Architecture

8. Testing

A System Test Plan will allow PSSM to gain confidence that KMAD understood PSSM's needs and requirements, and successfully implemented a useful solution to meet them. We propose a Testing Plan with intertwined and interrelated phases: Requirements Testing, Documentation Testing, Alpha Testing and Beta Testing. We provide a set of test cases below and make use of them during all four phases.

8.1 Requirements Testing

Requirements testing determines whether or not the solution meets the orginal business requirements. These will be performed by KMAD, using the test cases presented in section 8.5 below. This testing phase will occur in two major parts: 1) after each new system class is implemented, and 2) after the code for each major business process is committed. For example, after the Contract class is finished, test case #1 will be executed. Subsequently, after class TempEmployee is complete, test case #2 will be executed, etc. Similarly, after the code for reserving a candidate for a staffing request is committed, test case #5 will be executed.

8.2 Documentation Testing

Documentation testing determines whether or not the solution matches the documentation delivered by KMAD's documentation team. This documentation should be used by PSSM during hands-on training to familiarize staff with the use of the *TechStaff* application. This documentation will be delivered separately, but KMAD will test that the documentation matches the test cases below (8.5) immediately before Alpha testing begins, described below.

8.3 Alpha Testing

Alpha testing deterimes whether or not the solution meets the customer requirements by using test data. This is performed by PSSM staff, with the guidance of KMAD. Again, after each class and after each business process are implemented, the appropriate test case will be provided to PSSM staff for execution. Following the Agile/Scrum methodology, this puts the solution into the hands of the customer as quickly as possible, so that we can get feedback and make adjustments early on, if necessary.

8.4 Beta Testing

Beta testing is the same as alpha testing, except that it's performed with real data in a so-called "staging" system. This means that PSSM's existing spreadsheet system will be unaffected, and PSSM operations will be unchanged until the new TechStaff solution is accepted and approved for go-live. This testing will again be performed by PSSM staff, under KMAD supervision. After beta testing in the staging system is accepted by PSSM, we will go-live and designate the newly-created staging system as PSSM's production system moving forward.

8.5 Test Cases

As discussed, the following test cases will be used in all four testing phases outlined above.

Test case #: 1	*Test Case Name:* Login, logout; Reset password
System: TechStaff	*Subsystem:* Security
Designed by: Larry David	*Design Date:* 2016-11-15
Executed by: Cosmo Kramer	*Execution Date:* 2016-11-30
Short Description: Test the option code field during sales Invoice data entry.	

Pre-conditions:
The user has opened a web browser.

Step	*Action*	*Expected System Response*	*Pass/ Fail*	*Comment*
1	Enter the PSSM TechStaff URL in the browser.	The System displays the PSSM TechStaff web page. The user is prompted for a username and password.		
2	Enter the assigned username and password. Click the "Login" button.	The System displays the TechStaff home screen.		
3	At the home screen, click Logout.	The System displays the PSSM TechStaff web page. The user is prompted for a username and password.		
4	Below the login fields, click the "Forgot password?" link.	The System displays an input field labeled, "Please enter your email address to reset your password"		
5	Enter the new email address. Click "Reset my password".	The System sends an email to the user.		
6	The user checks their email and locates the email sent by TechStaff. The user clicks on the reset password URL included in the email.	The System opens a new browser window, connects to the TechStaff application, and displays two input fields, for entering and confirming the new password.		
7	Enter and confirm the new password. Click Save.	A message is displayed, "Your password has been successfully reset. You may		

		now log into the system.		
8	*Check post-condition 1.*			

Post-conditions:

1. The user is able to log into PSSM *TechStaff* with the newly-reset password.

Test case #: 2	*Test Case Name:* CRUD Contract
System: TechStaff	*Subsystem:* Contracts
Designed by: Wade Wilson	*Design Date:* 2016-11-15
Executed by: Bruce Wayne	*Execution Date:* 2016-11-30
Short Description: Test the Creation, Reading, Updating, and Deleting of a client Contract	

Pre-conditions:

The Customer has established a contract with a client.
The Customer has logged into the the TechStaff web application.

Step	*Action*	*Expected System Response*	*Pass/ Fail*	*Comment*
1	Test Create: Click the "Create Contract" button	The system presents a list of input fields. Fields surrounded by a red box are required.		
2	Fill in required Contract fields: ID, Expiration Date, Company Name, Address, Phone, Terms	The system enables the Save button at the bottom of the input fields		
3	Click Save.	The system presents a pop-up message "Contract #<ContractID>" has been saved. The list of existing contracts is displayed.		
4	*Check post-condition 1.*			
5	Test Read: Click the "Find Contract" button	The system presents a list of Contract search input fields: ID, Expiration Date, Name, Phone		
6	Type the desired Contract ID into the "ID" field. Click "Search"	The system presents a list of matching contracts.		
7	Select the desired Contract and click the "Open" button	The system displays the Contract details. The fields are read-only.		
8	*Repeat Test Read (5) for Expiration*			

	Date, Name, and Phone.			
9	Test Update: Open a Contract as described in Test Read (5) above.	The system displays the Contract details. The fields are read-only.		
10	Click the “Edit” button	The Contract fields become editable, except for the Contract ID field, which can’t be changed.		
11	Enter a change for the Phone field. Click Save.	The system presents a pop-up message “Contract #<ContractID>” has been saved.		
12	*Check post-condition 2.*			
13	*Repeat Test Update (9) for Expiration Date, Name, and Phone.*			
14	Test Delete: Open a Contract as described in Test Read (5) above.	The system displays the Contract details. The fields are read-only.		
15	Click the “Edit” button	The Contract fields become editable, except for the Contract ID field, which can’t be changed.		
16	Click the “Delete Contract” button	The system displays a pop-up message “Are you sure you want to delete Contract #<ContractID>? You cannot undo this action.” Two buttons are presented below: “Cancel” and “Delete”		
17	Click Delete.	The system displays a message, “Contract #<ContractID> has been deleted.”		
18	*Check post-condition 3.*			

Post-conditions:

1. The new Contract is saved in the database with the entered data.
2. The updated Contract is saved in the database with the updated data.
3. The Contract has been removed from the database.

Test case #: 3	*Test Case Name:* CRUD TempEmployee
System: TechStaff	*Subsystem:* Employee
Designed by: Barry Allen	*Design Date:* 2016-11-15
Executed by: Doctor Strangelove	*Execution Date:* 2016-11-30
Short Description: Test the Creation, Reading, Updating, and Deleting of a client Contract	

Pre-conditions:
The Customer has established a new relationship with a staff candidate. The Customer has logged into the the TechStaff web application.

Step	*Action*	*Expected System Response*	*Pass/ Fail*	*Comment*
1	Test Create: Click the "Create Employee" button	The system presents a list of input fields. Fields surrounded by a red box are required.		
2	Fill in required Employee fields: ID, Name, Address, Phone, Status, Years of Experience, Degree, and Expected Pay	The system enables the Save button at the bottom of the input fields		
3	Click Save.	The system presents a pop-up message "Employee #<EmployeeID>" has been saved. The list of existing contracts is displayed.		
4	*Check post-condition 1.*			
5	Test Read: Click the "Find Employee" button	The system presents a list of Employee search input fields: ID, Name, Phone, Status, Years of Experience, Degree, Expected Pay		
6	Type the desired Employee ID into the "ID" field. Click "Search"	The system presents a list of matching employees.		
7	Select the desired Employee and click the "Open" button	The system displays the Employee details. The fields are read-only.		
8	*Repeat Test Read (5) for Name, , Phone, Status, Years of Experience, Degree, and Expected Pay*			
9	Test Update: Open an Employee as described in Test Read (5) above.	The system displays the Employee details. The fields are read-only.		
10	Click the "Edit" button	The Employee fields become editable, except for the Contract ID and Name fields, which can't be changed.		
11	Enter a change for the Phone field. Click Save.	The system presents a pop-up message "Employee #<EmployeeID>" has been saved.		
12	*Check post-condition 2.*			
13	*Repeat Test Update (9) for Name, Address, Phone, Status, Years of Experience, Degree, and Expected Pay*			

14	Test Delete: Open an Employee as described in Test Read (5) above.	The system displays the Employee details. The fields are read-only.		
15	Click the "Edit" button	The Employee fields become editable, except for the Employee ID and Name fields, which can't be changed.		
16	Click the "Delete Employee" button	The system displays a pop-up message "Are you sure you want to delete Employee #<EmployeeID>? You cannot undo this action." Two buttons are presented below: "Cancel" and "Delete"		
17	Click Delete.	The system displays a message, "Employee #<EmployeeID> has been deleted."		
18	*Check post-condition 3.*			

Post-conditions:

1. The new TempEmployee is saved in the database with the entered data.
2. The new TempEmployee is saved in the database with the updated data.
3. The TempEmployee has been removed from the database.

Test case #: 4	*Test Case Name:* CRUD StaffingRequest
System: TechStaff	*Subsystem:* Employee
Designed by: Larry David	*Design Date:* 2016-11-15
Executed by: George Costanza	*Execution Date:* 2016-11-30

Short Description: Test the Creation, Reading, Updating, and Deleting of a StaffingRequest

Pre-conditions:

The Customer has received a temporary staffing demand from a Client.

The Customer has access the PSSM TechStaff webapp URL via web browser.

Step	*Action*	*Expected System Response*	*Pass/ Fail*	*Comment*
1	Test Create: Click the "Open Staffing Request" button	The system presents a list of input fields. Fields surrounded by a red box are required.		
2	Fill in required Staffing Request fields: ID, ContractID, Description,	The system enables the Save button at the bottom of the input fields		

	Experience Required, Degree Required, Pay Offered			
3	Click Save.	The system presents a pop-up message “Staffing Request #<RequestID>” has been saved. The list of existing Staffing Requests is displayed.		
4	*Check post-condition 1.*			
5	Test Read: Click the “Find Staffing Request” button	The system presents a list of Staffing Request search input fields: Request ID, Status, ContractID, Experience, Degree, Pay, Reserved Employee, Placed Employee		
6	Type the desired Staffing Request ID into the “Request ID” field. Click “Search”	The system presents a list of matching Staffing Requests.		
7	Select the desired Staffing Request and click the “Open” button	The system displays the Staffing Request details. The fields are read-only.		
8	*Repeat Test Read (5) for Status, ContractID, Experience, Degree, Pay, Reserved Employee, Placed Employee*			
9	Test Update: Open a Staffing Request as described in Test Read (5) above.	The system displays the Staffing Request details. The fields are read-only.		
10	Click the “Edit” button	The Staffing Request fields become editable, except for the Request ID field, which can’t be changed.		
11	Enter a change for the Pay Offer field. Click Save.	The system presents a pop-up message “Staffing Request #<RequestID>” has been saved.		
12	*Check post-condition 2.*			
13	*Repeat Test Update (9) for Status, Description, Experience Required, Degree Required, Pay Offered*			
14	Test Delete: Open a Staffing Request as described in Test Read (5) above.	The system displays the Staffing Request details. The fields are read-only.		
15	Click the “Edit” button	The Staffing Request fields become editable, except for the Employee ID and Name fields, which can’t be changed.		
16	Click the “Delete Employee” button	The system displays a pop-up message		

		"Are you sure you want to delete Staffing Request #<RequestID>? You cannot undo this action." Two buttons are presented below: "Cancel" and "Delete"		
	Click Delete.	The system displays a message, "Staffing Request #<RequestID> has been deleted."		

Post-conditions:
1. The new StaffingRequest is saved in the database with the entered data.
2. The new StaffingRequest is saved in the database with the updated data.
3. The StaffingRequest is deleted from the database.

Test case #: 5	*Test Case Name:* Reserve, place, close a Staffing Request.
System: TechStaff	*Subsystem:* Staffing Request
Designed by: Elaine Benes	*Design Date:* 2016-11-15
Executed by: Banya	*Execution Date:* 2016-11-30

Short Description: Test the candidate reservation, employee placement, and close functions of a Staffing Request.

Pre-conditions:
The Customer has received a temporary staffing demand from a Client.
The Customer has access the PSSM TechStaff webapp URL via web browser.
Based on a Contract search, the demand is determined to be valid.
Based on an Employee search, a suitable candidate is available.

Step	*Action*	*Expected System Response*	*Pass/ Fail*	*Comment*
1	Test Reserve: Click the "Reserve Candidate" button	The system presents a list of Employee as well as Staffing Request search input fields: Employee ID, Request ID, Employee Status, Request Status, ContractID, Employee Name, Employee Status, Expected Pay, Pay Offered, etc.		
2	Fill in desired search fields, e.g. Employee Name or Request ID	The system displays a list of items that match. For example, any employees matching the name OR any requests matching the ID are displayed.		

3	Select the desired Staffing Request and click "Candidates"	A list of available candidates is presented.		
4	Select a qualified and suitable candidate and click "Reserve."	A message is displayed: "Candidate <EmployeeID> has been reserved for Staffing Request <RequestID>." Arrangements department has been notified.		
5	*Check post-condition 1.*			
6	*Repeat Test Reserve (1, 2) as described above.*			
7	Select the desired Candidate and click "Staffing Requests"	A list of open Staffing Requests is presented.		
8	Select an appropriate Staffing Request and click "Reserve"	A message is displayed: "Candidate <EmployeeID> has been reserved for Staffing Request <RequestID>. Arrangements department has been notified."		
9	*Check post-condition 2.*			
10	Click the "Edit" button	The Staffing Request fields become editable, except for the Request ID field, which can't be changed.		
11	Enter a change for the Pay Offer field. Click Save.	The system presents a pop-up message "Staffing Request #<RequestID>" has been saved.		
12	*Check post-condition 2.*			
13	Test Place: From the TechStaff home screen, click "Place Employee"	A list of reserved candidates is listed.		
14	Select the desired candidate and click "Staffing Requests"	The system displays a list of open Staffing Requests.		
15	Select an appropriate Staffing Request and click the "Place" button.	A message is displayed: "Employee <EmployeeID> has been placed for Staffing Request <RequestID>, Contract <ContractID>." The Contract Manager is notified		
16	*Check post-condition 3.*			
17	From the TechStaff home screen, select "Search Staffing Requests"	A number of Staffing Request search fields are presented.		
18	Enter the Request ID for the placed Staffing Request from step 15 above.	The Staffing Request is displayed.		
19	Select the Staffing Request and choose "Close Request"	A message appears, "Staffing Request <RequestID> was filled and is now closed. The Client has been notified."		

	Check post-condition 4.			

Post-conditions:

1. The TempEmployee is marked as “reserved” in the database, and the associated Staffing Request now includes the EmployeeID in the ReservedEmployee field. A notification email regarding the candidate reservation was received by Arrangements.
2. The TempEmployee is marked as “reserved” in the database, and the associated Staffing Request now includes the EmployeeID in the ReservedEmployee field. A notification email regarding the candidate reservation was received by Arrangements.
3. The TempEmployee, Staffing Request, and Contract records have been updated in the database. The billing department was notified to generate a bill for the client. A copy of the Staffing Request was emailed to the Client.
4. A notification email was sent to the Client indicating that the Staffing Request was closed.

9. Change Management

10. Conclusion